Autonomous lecture recording with a PTZ camera while complying with cinematographic rules

Dries Hulens, Toon Goedemé
KU Leuven, EAVISE
Sint-Katelijne-Waver, Belgium
Email: dries.hulens@kuleuven.be, toon.goedeme@kuleuven.be

Tom Rumes
Thomas More, PRO media lab
Mechelen, Belgium
Email: tom.rumes@thomasmore.be

***Abstract*—Nowadays, many lectures and presentations are recorded and broadcasted for teleteaching applications. When no human camera crew is present, the most obvious choice is for static cameras. In order to enhance the viewing experience, more advanced systems automatically track and steer the camera towards the lecturer. In this paper we propose an even more advanced system that tracks the lecturer while taking cinematographic rules into account. On top of that, the lecturer can be filmed in different types of shots. Our system is able to detect and track the position of the lecturer, even with non-static backgrounds and in difficult illumination. We developed an action axis determination system, needed to apply cinematographic rules and to steer the Pan-Tilt-Zoom (PTZ) camera towards the lecturer.**

***Keywords*-automatic camera system; cinematographic rules; computer vision; tracking; real-time;**

I. Introduction

There is a significant difference between being present at a lecture and watching one on a screen that is recorded with a static camera. When you are present at the lecture, you can follow the speaker and focus on what he tells and does. This is much more interactive compared to watching the lecture on a screen, where your attention is quickly lost. However, when we look at, for example a TED talk[1], their way of filming makes the presentation much more interesting to watch since they work with a professional camera crew and a director who takes a set of cinematographic rules into account.

Our goal is to reach the same viewing experience while watching lectures recorded by an automated system. To accomplish this, we developed an automatic camera man (PTZ camera-unit) that is able to:

- Detect and track a single person/lecturer
- Change between different types of shots
- Listen to high-level instruction from a virtual or human director
- Take cinematographic rules into account
- Work in Real-Time

[1]Presentations filmed by a professional camera crew and director. http://www.ted.com/talks?lang=en

Figure 1. Example of the *rule of thirds* where the (anonymous) actor is framed on a third of the image, leaving some empty image space for the action. Also the *head room* rule is taken into account, the point of intrest (eyes) are framed on a third under the top of the image.

By tracking the lecturer we can make sure that he is framed well in the picture at any moment and viewers can't be distracted, e.g. by things happening in the audience. The director can ask for different shots like a medium-shot to draw the attention on the lecturer or a long-shot of the room to give an overview. The director still decides which virtual camera should take which shot, and the type of shot command is passed to the virtual camera man. By doing this, the setup is easily expandable with different camera-units to record from different angles and shots simultaneously.

Our system has many advantages because it works in difficult situations with dynamic backgrounds and changing illumination. The camera units can easily be installed in different rooms without calibration. And more important, the images are aesthetically more interesting to watch because we take the cinematographic rules into account.

This paper is structured as follows: we first describe the most common cinematographic rules in section II. Section III gives an overview of related work. In section IV we present our approach, while in section V we reveal our experiments and results. Finally, we end with a conclusion

in section VI.

II. Basic cinematographic rules

A human camera man assures continuously an aesthetical shot composition by panning, tilting and zooming the camera. For this, the camera man has to apply some cinematographic rules. To begin with, the camera man should make sure that motions are smooth and logical and that the shot composition is well chosen. The appropriate zoom setting should be selected for every shot as well as the correct position of an actor in the frame. Remarkably, the actor focused on should not be centered in the image. It is aesthetically much more interesting if the actor is pictured to the left or right of the image, leaving a bit of empty image space where the action takes place. A person looking e.g. to the right, should be pictured on the left side of the image. Or, an image of a skier going down a slope, should leave some empty image space next to the skier, at the place where the skier is skiing to. This rule is called the *rule of thirds*, an example can be seen in Figure 1. Another rule a human camera man applies is holding the actor vertically centered in the image with a minimum of space between the head and top borders of the image. This space is called the *head room*. Actually, the point of interest (in the case of a medium shot, the eyes) is helt $1/3th$ under the top of the image. It is also obvious that the camera man will not zoom in or out while the image of his camera is used, except when the director asks this explicitly to change the type of shot. In this paper we demonstrate an automatic solution for two types of shots, namely long shot, where the whole person is visible in the image and medium shot, where the person's head and shoulders are visible.

In our approach we implemented these basic rules so that the movement of the camera will be smooth, the *rule of thirds* and *head room* is respected and the camera only zooms in and out to change the type of shot or when the director asks.

Another famous basic cinematographic rule is that the action axis may not be crossed when changing the camera viewpoint (the so-called $180\,^{\circ}$-rule). With the PTZ cameras used in this paper, which have a fixed viewpoint, we can not take this rule into account. Nevertheless, in future work where our PTZ cameras will become more mobile, this rule will also be implemented.

III. Related work

A lot of research is already done in camera automation for recording lectures. In [1] a system is developed to track a lecturer with a PTZ camera. The system detects the lecturer by skin color and applies simple cinematographic rules (on the level of the movie director), e.g. switching frequently between shots and choosing a good duration for the shot. To locate the audience and the speaker they use an external positioning system. This is an enormous disadvantage because in every new room the system should be recalibrated. This is also the case in [2] where they use a depth sensor to track the lecturer and where they don't apply important cinematographic rules. Another example where they do apply cinematographic rules is in [3]. Here they use a network of static cameras to record a presentation or lecture. The disadvantage of this system is that they need several cameras to cover the room where we only need a single PTZ camera.

Our approach differs from previous methods because we use more reliable object detection techniques and we use only one camera unit (but can expand the system easily with more camera units). There is no calibration needed, thus the system can be placed in every room. A related approach is [4] where they detect the lecturer and the projection screen. The lecturer is detected with a face detection algorithm using the Viola and Jones [5] technique. They also determine the action axis by estimating the gaze direction. This is done by assuming that when one looks to the right, the right part in the image of the head (nose, lips, eyes, cheek, ...), contains more skin color than the left part (hair, ear.). This is true but also has to be calibrated for every skin color, and depends on the illumination conditions. Because we use different models of a person's head to estimate the gaze orientation, our system is not dependent on the illumination conditions and skin color and therefore more robust. Their method is also just limited to a medium-shot because they only use a face detection technique. We are not restricted to one type of shot because we use multiple detection techniques in our approach.

An important aspect of each automatic lecture recording system is the technique used to detect the lecturer's position. One of the simplest techniques to detect a moving object such as a lecturer is *background subtraction*. When using a static background, the background can be subtracted from the image, whereby the moving foreground object pops out. This technique is used in [6] to determine the direction of movement of a lecturer. Another technique is color segmentation. This is used to filter-out skin color, as mentioned above, to detect a person or a person's head. These techniques work well under perfect conditions, but fail when working with a dynamic background, or changing illumination (like in real life). Therefore more reliable techniques are needed, which we can find in the object detection literature. For face detection, there are two well-known techniques namely the Haar/AdaBoost-based face detection algorithm of Viola and Jones [5], as mentioned before, and the Local Binary Patterns (LBP) technique proposed in [7].

For full person detection, other techniques are used. Initially Dalal and Triggs [8] proposed the *Histogram of Oriented Gradients* (HOG) pedestrian model for human detection, where an image (on different scales) is compared with a model of a person. Felzenszwalb *et al*. extended this idea to a part-based HOG model [9], to improve the

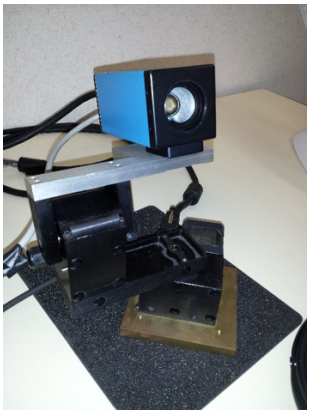

Figure 2. Camera unit: a zoom camera mounted on a pan-tilt unit.

accuracy. Of course, this leads to lower processing speeds, because the complexity of the model increases. To speed-up detection, Benenson *et al.* [10] rescaled their model instead of the image. An even faster approach is presented in [11] where they exploit scene constraints combined with a CPU/GPU implementation of their algorithm.

IV. Approach

Our goal is the development of an autonomous camera unit to record lectures without the need of a human camera man. This autonomous system is automatically tracking the lecturer and follows his movements. The type of shot is still decided by a director, which can be human or virtual, and is in this paper limited to medium shot and long shot. Our system consists of a camera with a motorized zoom mounted on a pan-tilt system to turn the camera towards the lecturer, as can be seen in Figure 2. This system is easily expandable with more camera units since they are autonomous and only need high level instructions (type of shot) of a director.

We split up this task in three steps. In the first step the lecturer's actual position is determined and tracked using a Kalman filter. Subsequently the cinematographic rules are applied so that the desired position of the lecturer in the image is known. In the last step, the camera is steered to the lecturer in an image-based visual servoing approach. The total algorithm is pictured in a block diagram in Figure 3.

A. Lecturer detection

First we need determine the current position of the lecturer. As we explained in section III, the easiest way is background subtraction to abstract the lecturer from the scene. Because we have a dynamic background, since our camera is moving along with the lecturer, this is not an option. Therefore we rely on object detection techniques to cope with these dynamic backgrounds. We use two types of object detectors depending on the type of shot. For a long shot, where the whole person is visible, we use a pedestrian detection technique based on HOG/SVM while

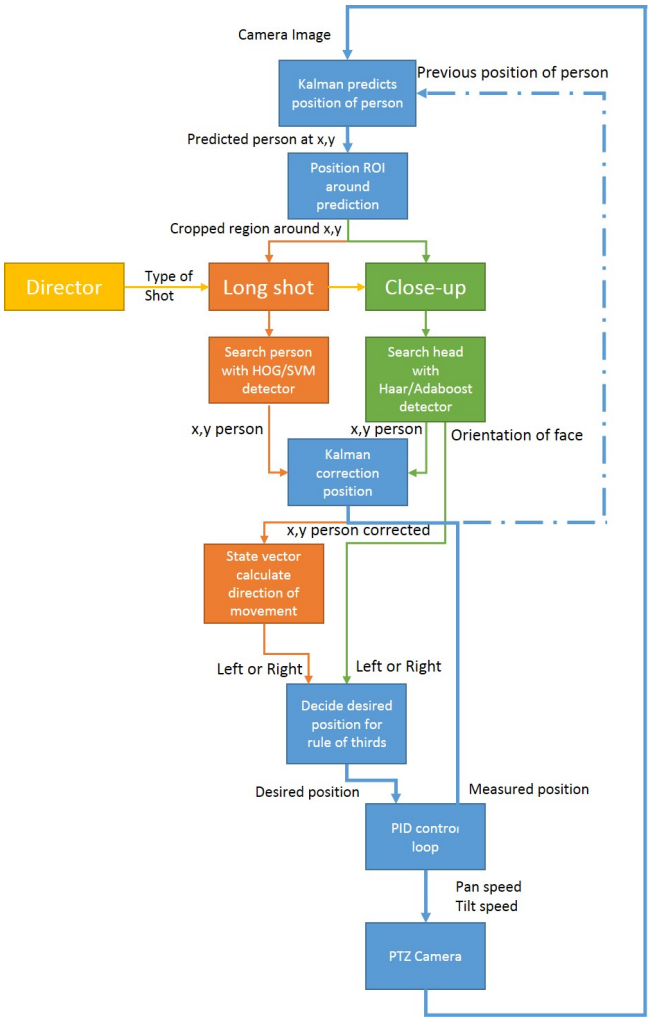

Figure 3. Block diagram of algorithm. The yellow color is the director, Green is for a medium-shot, Orange is for a long-shot and Blue is common.

for a medium shot or close-up we use a Haar/AdaBoost-based face detection technique. Both methods are selected because of their excellent computation speed, as mentioned in section V. HOG/SVM is an object detection technique of Dalal and Triggs [8] that builds on an object class model based on *Histogram of Oriented Gradients* (HOG) image features from a large number of training images. We used a full-body model for the detection of actors in a long shot. This method works fine when the whole person or lecturer is visible in the image.

When the director requests a medium shot of the lecturer, this method is no longer suitable because only his head is visible. In this case we use the Haar/AdaBoost-based face detection algorithm of Viola and Jones [5]. This detector is more accurate than LBP and is still fast enough to perform face detection in real-time. The detector thus depends on the type of shot that is desired. With these two detectors the most common shots are covered. These detectors return the

Figure 4. Illustration of *rule of thirds* on actual TED footage. The rectangles are the result of what we want to get with our person and face detection. The arrow represents the action axis.

position, height and reliability score of their detection (face or person) that are found in the image. To track the position of the lecturer over time, we can use a Kalman filter [12] with the following state vector and update matrix, assuming a constant velocity motion:

$$\mathbf{x} = \begin{bmatrix} x_{im} \\ y_{im} \\ v_{x,im} \\ v_{y,im} \end{bmatrix} \qquad A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

where x_{im} and y_{im} are the position of the object on the image plane and $v_{x,im}$ and $v_{y,im}$ are the velocity of the object on the image plane. The Kalman filter predicts the position where the lecturer should be in the frame, which is needed when the lecturer is not detected (e.g. when occluded behind his desk). A second advantage of using a Kalman filter is that the noise on the measured position of the lecturer is being filtered out.

Because the position of the lecturer is being tracked by the Kalman filter, and the next position is predicted, we can search the lecturer in a much smaller region around this prediction. A first detection is performed on the entire frame. In subsequent frames we only search in this smaller region of which the size is determined by the size of the previous detection. If the detection is lost for multiple frames in a row (10 frames in our experiments), we again search over the entire image. This accelerates the person detection algorithm significantly as explained in section V. This acceleration is a big advantage because, in the future, we want the camera units to be fully autonomous with an embedded processor and still performing the person detection in real time. Of course, this acceleration technique only works when the cropped region is smaller than the original frame. When this is not the case, e.g. when the detection is larger, an acceleration can also be achieved by downsampling the original image.

At this point we know the actual position of the lecturer. In the next step we will determine the desired shot position depending on the cinematographic rules.

B. Complying with Cinematographic Rules

By taking cinematographic rules into account, the viewing experience becomes much more attractive and improves the focus on the lecturer. A first rule we implemented is the *rule of thirds*. As mentioned before, in professional recordings (e.g. a TED talk), the person of interest is never in the middle of the image but always on $1/3th$ left or right in the image, depending where the action is taking place.

To know where the action is taking place, we first have to find the action axis. The action axis goes through the person of interest and points towards the action, as can be seen in Figure 4. The action axis can be found in two ways, by detecting the person's gaze orientation or by finding the direction of motion of the person. In this paper we use the gaze orientation on medium shots and motion direction on long shots to determine the action axis. In the future, we will combine them for an even more reliable action axis estimation.

1) Gaze orientation: Determining the gaze orientation is accomplished by running the face detection algorithm with two different models. We use a frontal model and profile model (face looking left) of the face. To detect a face oriented to the right, the image is mirrored around the *y-axis*. The face detector unfortunately does not return a confidence score, only the number of detections and the coordinates of the detections, so an extra step is needed. To get a confidence score nonetheless, we run the models on every scale without non-maximum suppression (grouping detections in the same region). We use this number of co-occuring detections as our score. Suppose we have a face oriented to the right, the right-profile model will give us more detections than the left and frontal model. At this point we do not know which of the detections is our face (because we have true and false detections), but we know which model has the highest score. By running the model with the highest score (in our example the right-oriented model) a second time over the image, but with non-maximum suppression, we find the coordinates of the face. In effect, four models are tested over the image. This is not a problem because of the high processing speed of the algorithm.

2) Motion direction: Determining the direction of movement is not as straight forward as the gaze orientation. For this, we have to find the velocity (the sign of the velocity gives the direction) of the person of interest in the world, $\mathbf{v_p}$. Unfortunately, because the PTZ camera can rotate, we only observe a relative motion of the person in the image $\mathbf{v_{im}}$. To calculate the real world velocity we have extended the Kalman filter from equation 1 in section IV-A, such that $\mathbf{v_p}$ can be calculated. We observe that $\mathbf{v_p}$ depends on $\mathbf{v_{im}}$ and the velocity of the camera $\mathbf{v_c}$ (figure 5):

$$\mathbf{v_p} = \mathbf{v_{im}} - \mathbf{v_c} \tag{2}$$

The (magnitude of the) velocity of the camera in the world v_c can be calculated with:

$$v_c = \omega_c \beta R \tag{3}$$

where ω_c is the angular velocity of the camera and R is the distance. ω_c is known because it is the pan command

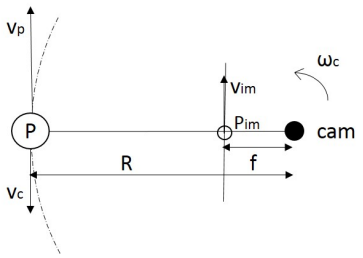

Figure 5. Top-down view of person P moving in front of a camera. The apparent motion $\mathbf{v_{im}}$ of the person observed in the image gives only a relative indication of the real motion of the person in the world $\mathbf{v_p}$ because of the rotation of the camera ω_c.

our system sends to the PTZ unit. ω_c is multiplied by a calibration factor β so we can express v_c in $pixels/frame$ instead of m/s. To find β we panned the camera with a constant angular velocity and measured the shift of a fixed object in the image plane. The distance between the camera and the person R is not known, but can be approximated based on the measured height h of the person or face in the image plane. The smaller the object appears in the image, the further away the person is, as can seen in the next equation:

$$R = \frac{\alpha}{h} \tag{4}$$

α is to be calibrated for every type of shot, this should be done only once by measuring the height of a person at a certain distance R, on the image plane. The accuracy of α and β is not so important because we only need the direction of the person's velocity and not the exact value. Combining equation 3 and 4 yields the following relation:

$$v_c = \omega_c \frac{\alpha\beta}{h} \tag{5}$$

As (2) indicates, $\mathbf{v_{im}}$ is needed to determine $\mathbf{v_p}$. However $\mathbf{v_{im}}$ is easily extracted from the Kalman state vector $\mathbf{x}$ calculated in IV-A. By inserting $\mathbf{v_p}$ and $\mathbf{v_c}$ in $\mathbf{x}$, the unknown person world speed $\mathbf{v_p}$ can also be calculated by the Kalman fiter with the advantage that $\mathbf{v_p}$ is being tracked and thereby noise is filtered out. To accomplish this, $\mathbf{x}$, A and H from equation 1 have to be extended to calculate (2). We choose $\mathbf{x}$, A and H to be

$$\mathbf{x} = \begin{bmatrix} x_{im} \\ y_{im} \\ v_{x,im} \\ v_{y,im} \\ v_p \\ v_c \end{bmatrix} \qquad A = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{6}$$

$$\mathbf{z} = \begin{bmatrix} x_{im} \\ y_{im} \\ v_c \end{bmatrix} \qquad H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The observation vector $\mathbf{z}$ contains, beside the position of the detected person (x_{im}, y_{im}) also v_c, calculated with equation 5 for each frame.

Since we now know the direction of movement $\mathbf{v_p}$ or the gaze estimation, we can decide the desired position of the person in the image. If the person is moving or watching to the left, our camera system should place the person on the right of the image, at $2/3th$ of the x-axis, as can be seen in Figures 7 and 8. Of course, the gaze estimation and direction of movement are filtered so that the shot position does not change when the person is looking to the other side for a short period of time. This filtering is accomplished by inserting the direction (*L* or *R*) in a fifo from a certain length. Only if every value in the fifo is equal, the desired position is adapted.

A second rule that we implemented is the *head room*. This is the vertically desired position (*y-axis*) of a person, where the point of interest should be located $1/3th$ under the top of the image. When recording a medium shot the point of interest are the eyes, for a long shot the point of interest is the head. We derived the position of the eyes as a relative position w.r.t. the height of the face. For the face model we used, the eyes are located 6% above the center of the head. In a long shot the point of interest is the head. For the HOG person detector, the head is located 34% above the center.

At this point we know the actual and desired position of the person of interest, and the camera can be steered to the desired position. How this is performed, is explained in the next subsection.

C. Image-Based Visual Servoing

To turn the camera towards the person of interest, we are using an image-based visual servoing (IBVS) [13] method. In contrast to position-based visual servoing (PBVS), we do not calculate an exact angle for the camera to turn to, but we calculate the direction and speed with which the camera should turn towards the desired position. This allows to adjust the speed at every frame, making it possible to turn the camera at a much higher speed when the distance between desired position and actual position is large and slow down when the distance becomes smaller.

To move the camera we are using a Pan-Tilt unit as seen in Figure 2. This unit can be used in speed- or position mode. Because we are using IBVS the unit is used in speed mode. The speed for the x- and y-axis are calculated in a PID control loop, and is depending on the error between the desired and actual position. By using a PID control loop, the camera turns very smooth which is advantageous for recording.

V. Experimental results

To evaluate our approach we attached a zoomable FireWire camera (ImagingSource DBK 21BF04-Z2) on a pan-tilt unit (PTU D46-17 from FLIR) as shown in Figure 2. The pan-tilt unit is controlled in speed mode, using serial commands. We first implemented the HOG object detector

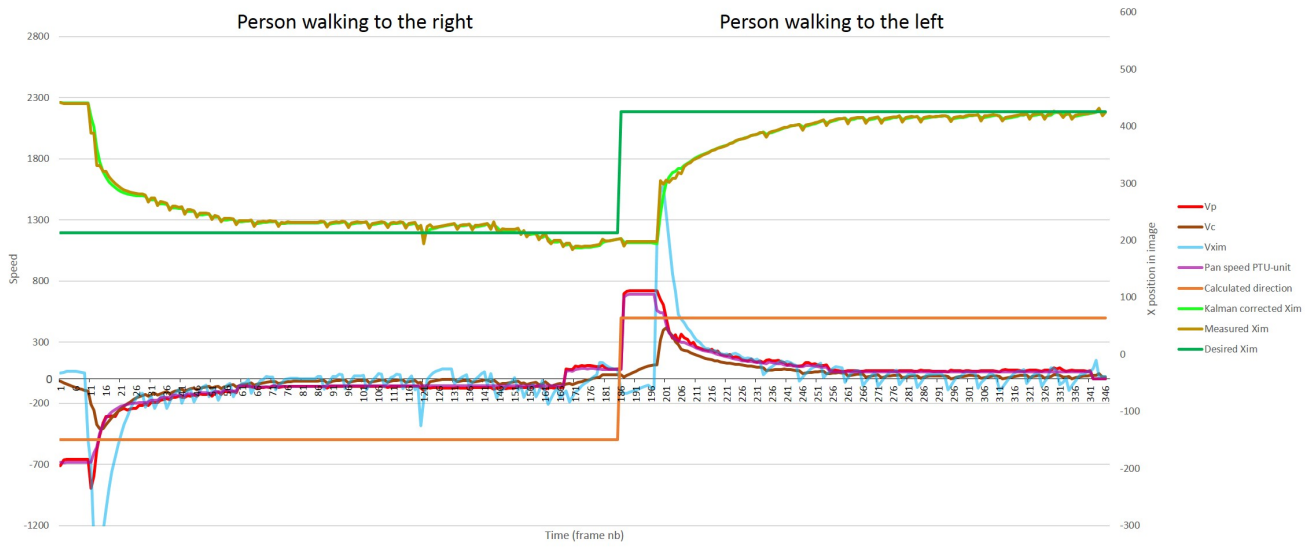

Figure 6. Illustration of the position and velocity measurements of our system on one tracking sequence. The top three graphs represent the measured position (light brown), the Kalman fitered position (light green) and the desired position (dark green) of the detected person. The bottom graphs represent the calculated direction (orange), the pan-speed sent to the pan-tilt unit (purple), the speed of the person in the image x_{im} (blue), the speed of the camera in the world V_c (brown) and the calculated speed of the person in the world V_p (red). A video can be found on *http://youtu.be/U4dzUqUZ_sk*.

Figure 7. Result of our algorithm when a long shot is desired. The method to detect the lecturer is HOG person detection. The body is detected and indicated with a pink rectangle. The action axis is determined and displayed in red. The desired position, taking the *rule of thirds* and the *head room* into account, is displayed with green lines. The top three images show a person walking to the right, the bottom three images show a person walking to the left.

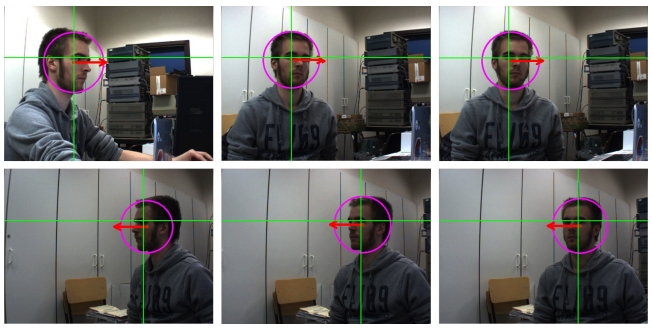

Figure 8. Result of our algorithm when a medium shot is desired. The method to detect the lecturer is Haar/Adaboost face detection. The face is detected and indicated with a pink ellips. The action axis is determined and displayed in red. The desired position, taking the *rule of thirds* and the *head room* into account, is displayed with green lines. A video can be found on *http://youtu.be/C9qRWiFTKEo*.

to determine the actual position of the lecturer in a long shot. Out of the box, the algorithm runs at about $17fps$ on images of 640×480 pixels, on a standard desktop PC (I7@3GHz). We expanded the algorithm by predicting the position of the lecturer in every frame with a Kalman filter. Around the predicted position a search region is cropped (based on the height of the person), thereby eliminating the need to perform a full image search. For example, with a cropped region of 320×240 pixels, a speed up of $\times4$ (or $68fps$) is achieved. Next we determined the action axis based on the direction of movement. We use this action axis to determine the most optimal position of the person in the image according to the *rule of thirds*. The point of interest is calculated using the detection coordinates of the face or body, to insert the *head room*. The actual and desired position are fed to a PID control loop that rotates the camera to track the lecturer while complying with the cinematographic rules. The result of the algorithm for a long shot is demonstrated in Figure 7.

In order to perform ground thruth-based verification of the developed system, we mounted a Barbie doll on a motorized translation table. Figure 6 shows the measured velocities and positions (in the x-direction) when tracking the doll ”walking” to the right, turning at frame 185 and returning to the left. The top three graphs indicate the measured position, the Kalman-filtered position and the desired position of the

person. As prescribed by the *rule of thirds*, when walking to the right or left, the desired position is at respectively $1/3th$ and $2/3th$ of the image width. As can be seen, the measured position gradually reaches the desired position. When the doll changes direction the position remains constant for several frames. This is because no detection is found (due to motion blur) when the camera rotates from $1/3th$ to $2/3th$. The bottom graphs represent the calculated direction, the pan-speed sent to the pan-tilt unit, the speed of the person in the image x_{im}, the speed of the camera in the world v_c and the calculated speed of the person in the world v_p. Since in our system we control the camera in such a way that the person remains at a fixed position in the image ($1/3th$ or $2/3th$ depending on the action axis), ideally $v_{x,im}$ should be zero (when walking at fixed speed - see e.g. frame 80-160). Some noise is noticeable since in practice it is impossible to always keep the person at the exact desired position. A similar observation is seen on v_c, since this speed depends on the detected height (see equation 5), which slightly varies for each detection. However, in the stabilized situations (frame 80-160 and frame 230-350) v_p remains exactly constant because the translation table moves the doll at a constant speed in the world.

When a medium shot is requested by the director, HOG can not be used since only the face and shoulders are visible. Therefore we implemented a Haar/AdaBoost face detection algorithm. The same methodology as explained above is also used for this algorithm, resulting in a speed-up of $\times 6$ (or $102fps$). Here, the action axis is found by detecting the face orientation as can be seen in Figure 8.

VI. Conclusion and future work

We developed a system to track a single lecturer with a PTZ camera based on two object detection techniques, namely person detection and face detection, making our approach very reliable. The main novelty is that we take cinematographic rules into account, which ensures that the viewer remains focused and the viewing experience is aesthetically more interesting. The action axis is determined by calculating the direction of movement and the gaze orientation. A PID control loop ensures smooth movement of the camera. Because of the speed of the algorithm it will be easy to downscale for embedded hardware and still perform the calculations in real-time. In the future we want to combine the direction of movement and the gaze orientation to make an even better action axis determination system. We also want to add other models for other shots: close-up, torso,... A solution should be found to retrieve the person or lecturer when falling out of the field of view. We also plan to extend the system to 3D where we can find the x, y and z position of the person, to use an Unmanned Aerial Vehicle (UAV) as a virtual autonomous camera man.

References

[1] F. Lampi, S. Kopf, M. Benz, and W. Effelsberg, "A virtual camera team for lecture recording," *Multimedia, IEEE*, vol. 15, no. 3, pp. 58–61, 2008.

[2] M. B. Winkler, K. M. Hover, A. Hadjakos, and M. Muhlhauser, "Automatic camera control for tracking a presenter during a talk," in *Multimedia (ISM), 2012 IEEE International Symposium on*. IEEE, 2012, pp. 471–476.

[3] P. Doubek, I. Geys, T. Svoboda, and L. Van Gool, "Cinematographic rules applied to a camera network," in *Omnivis2004: The fifth Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras*. Prague, Czech Republic: Czech Technical University, 2004, pp. 17–29.

[4] H.-P. Chou, J.-M. Wang, C.-S. Fuh, S.-C. Lin, and S.-W. Chen, "Automated lecture recording system," in *System Science and Engineering (ICSSE), 2010 International Conference on*. IEEE, 2010, pp. 167–172.

[5] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *CVPR*, 2001.

[6] Q. Liu, Y. Rui, A. Gupta, and J. J. Cadiz, "Automating camera management for lecture room environments," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '01. New York, NY, USA: ACM, 2001, pp. 442–449.

[7] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 971–987, 2002.

[8] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005, pp. 886–893.

[9] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.

[10] R. Benenson, M. Mathias, R. Timofte, and L. Van Gool, "Pedestrian detection at 100 frames per second," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2903–2910.

[11] F. De Smedt, K. Van Beeck, T. Tuytelaars, and T. Goedemé, "Pedestrian detection at warp speed: Exceeding 500 detections per second," in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, ser. CVPRW '13. IEEE Computer Society, 2013, pp. 622–628.

[12] G. Welch and G. Bishop, "An introduction to the kalman filter," University of North Carolina at Chapel Hill, Chapel Hill, NC, USA, Tech. Rep. 95-041, 1995.

[13] F. Chaumette and S. Hutchinson, "Visual servo control. i. basic approaches," *Robotics & Automation Magazine, IEEE*, vol. 13, no. 4, pp. 82–90, 2006.